CUSTOMER NO. 23932

Docket No.: 50886-00023USPX
(PATENT)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Matt Morris

Application No.: Not Yet Assigned

Filed: February 16, 2004 Art Unit: N/A

For: ROUTING OF DATA STREAMS Examiner: Not Yet Assigned

CLAIM FOR PRIORITY AND SUBMISSION OF DOCUMENTS

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

I hereby certify that this correspondence is being deposited with the U.S. Postal Service as Express Mail, Airbill No. EV 227 870 332 US, in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date shown below.

Dated: February 16, 2004 Signature: _____
(Margo Barbarash)

Dear Sir:

Applicant hereby claims priority under 35 U.S.C. 119 based on the following prior foreign application filed in the following foreign country on the date indicated:

Country	Application No.	Date
European Patent Office	03251091.9	February 24, 2003

In support of this claim, a certified copy of the said original foreign application is filed herewith.

Dated: February 16, 2004

Respectfully submitted,

By _____
André M. Szuwalski
Registration No.: 35,701
JENKENS & GILCHRIST, A PROFESSIONAL CORPORATION
1445 Ross Avenue, Suite 3200
Dallas, Texas 75202
(214) 855-4500
(214) 855-4300 (Fax)

Eur päisches Patentamt	European Patent Office	Office européen des brevets

Bescheinigung | Certificate | Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

Patentanmeldung Nr. | Patent application No. | Demande de brevet n°

03251091.9

Der Präsident des Europäischen Patentamts;
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

R C van Dijk

Anmeldung Nr:
Application no.: 03251091.9
Demande no:

Anmeldetag:
Date of filing: 24.02.03
Date de dépôt:

Anmelder/Applicant(s)/Demandeur(s):

STMicroelectronics, Ltd.
1000 Aztec West
Almondsbury,
Bristol, BS32 4SQ
GRANDE BRETAGNE

Bezeichnung der Erfindung/Title of the invention/Titre de l'invention:
(Falls die Bezeichnung der Erfindung nicht angegeben ist, siehe Beschreibung.
If no title is shown please refer to the description.
Si aucun titre n'est indiqué se referer à la description.)

Routing of data streams

In Anspruch genommene Priorität(en) / Priority(ies) claimed /Priorité(s) revendiquée(s)
Staat/Tag/Aktenzeichen/State/Date/File no./Pays/Date/Numéro de dépôt:

Internationale Patentklassifikation/International Patent Classification/
Classification internationale des brevets:

H04L12/00

Am Anmeldetag benannte Vertragstaaten/Contracting states designated at date of filing/Etats contractants désignées lors du dépôt:

AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IT LU MC NL PT SE SI SK TR LI

ROUTING OF DATA STREAMS

The present invention relates to the routing of data streams, and in particular streams constituted by data packets.

A transport stream multiplexer (TSMUX) has been implemented which can route a data stream received as an input to any one of a number of outputs. It is an aim of the present invention to expand the capabilities so that a number of input streams can be merged and can be directed to one or more of a plurality of output destinations.

According to one aspect of the present invention there is provided a stream routing unit for routing each of a plurality of input packet streams to any of a plurality of destinations, the stream routing unit comprising a plurality of input ports for receiving respective input streams; a plurality of output ports associated with respective destinations to which the input packet streams can be routed; storage means for holding packets of the input packet streams at addressable locations each identifiable by an address; and an assignment data structure identifying for each input stream the or each destination to which it is to be routed; and a packet allocation data structure holding for each new incoming packet a source identifier identifying the origin of the packet and the address in the storage means where the packet is held, together with information identifying the intended destination of the packet derived from the assignment data structure.

Another aspect of the invention provides a method of routing packet streams from a plurality of sources to any of a plurality of destinations, the method comprising: receiving said packet streams; identifying for each input packet stream the or each destination to which it is to be routed using an assignment data structure; holding each packet of the packet stream at an addressable location identifiable by an address in a storage means; holding for each new incoming packet a source identifier identifying the origin of the packet and the address in the storage means where the packet is held, together with information identifying the intended

destination of the packet derived from the assignment data structure, and using said information identifying the intended destination to route the packet from the storage means to the or each output port associated with the respective identified destination(s).

A further aspect of the invention provides a communications system which utilises a stream routing unit as hereinabove defined together with a plurality of sources for the input packet streams and destinations for receiving output streams.

As described below, the preferred embodiment of the invention provides an intellectual property (block of logic) known as TSmerger which merges multiple lower bit rate transport streams to a single higher bit rate transport stream for processing by a single programmable transport interface (PTI). For example, nine input streams can be merged into three output streams, with each input stream being able to be routed to any output stream, or to multiple destinations.

In the described embodiment, the TSmerger IP is implemented by storing packets of incoming streams in a single SRAM in a stream merger unit, which effectively behaves as a series of first-in first-out buffers (FIFOs).

For a better understanding of the present invention and to show how the same may be carried into effect, reference will now be made by way of example to the accompanying drawings in which:

Figure 1 is a schematic diagram of a communications system incorporating a stream routing unit;

Figure 2 illustrates a source to destination matrix;

Figure 3 illustrates a packet allocation table;

Figures 4A to 4D illustrate the assignment of destination pointers associated with the packet allocation table; and

Figures 5A to 5D illustrate the effect on the packet allocation table of adding and removing packets into and from the memory.

Figure 1 is a schematic diagram of a stream routing unit referred to herein as a TSmerger unit. The TSmerger unit is denoted generally by reference numeral 2 and has a plurality of input ports TSin1 ... TSin4 for receiving respectively each of a plurality of low bit rate input packet streams LBR1 ... LBR4. It will be appreciated that in practice any number of input streams may be present, four being illustrated by way of example only. Each input stream LBR1 ... LBR4 is derived from a respective source SRC1 ... SRC4. The TSmerger unit similarly has a plurality of output ports TSout0 ... TSout3, three of which are illustrated by way of example. These output ports are for the output of high bit rate output streams labelled HBR0 ... HBR2. Each output stream is supplied to a respective destination dest0, dest1, dest2 respectively. These destinations can take any suitable form, for example they can be programmable transport interfaces (PTI) which process the stream data, or they could be audio or video systems or anything capable of handling a data stream.

The TSmerger unit itself is capable of merging the lower bit rate transport streams to individual higher bit rate streams for processing by respective single PTIs. Each input stream can be routed to any output stream, and each input stream may be distributed to multiple output streams and thereby to multiple destinations.

The TSmerger unit 2 illustrated in Figure 1 includes a memory (SRAM) 3 for holding packets of the input streams, a packet allocation table 4 (PAT), a source to destination matrix 6 and a processing means 8 which implements an algorithm to control removal of packets from the SRAM 3.

The algorithm 8 controls the removal of packets from the SRAM 3 to the destinations dest0, dest1, dest2 in such a way as to maintain maximum bandwidth, while allowing any source stream to go to any destination. Packets from multiple sources are merged without breaking individual packets (i.e. streams are merged at the packet level and not at the byte level) and packets

from a single stream are read from the SRAM in the correct order, that is in the order in which they arrived.

To maintain maximum bandwidth, each packet is only read from the SRAM 3 once, so that if a packet from a particular input port is destined for more than one destination, it can only be output from the SRAM 3 when the ports for both of those destinations are free.

This is implemented as described in the following. The source to destination matrix 6 is a data structure illustrated in Figure 2. This matrix 6 maps sources to destinations and can be changed on the fly. The source numbers are indicated on the left hand side of the rows of the matrix, and the destinations are illustrated at the top of the columns of the matrix. A "1" in each square of the matrix indicates that that particular source is to be mapped to that particular destination. A "0" indicates that that source stream must not be mapped to that particular destination. As is clear from the matrix 6 in Figure 2, some source streams (from source 2 and source 4) are mapped to more than one destination.

Figure 3 illustrates the packet allocation table 4 which takes the form of a second data structure. This data structure takes the form of an array consisting of a plurality of slots SLOT1, SLOT2, etc, each slot containing a source identifier src_id of a particular packet in association with the address addr which is the start address of that packet in the SRAM 3. The source identifier is inserted into the packet header of each packet at the respective input port of the merger unit at which the packet is received. The source identifier insertion circuitry is labelled 7 in Figure 1. The packet allocation table 4 includes a write pointer wr_pointer and three destination pointers, dest0, dest1, dest2 each associated with a particular destination as illustrated in Figure 1. The pointers are implemented in any suitable known way. In Figure 3 the write pointer is shown pointing to the next available empty slot (SLOT5) in the packet allocation table 4. The destination pointers dest0, dest1, dest2 are shown pointing to the two top full slots of the

array (SLOTS 3, 4) illustrating the temporary assignment of those pointers when those slots of the array were just filled as will be described in more detail later.

Figures 4A to 4D illustrate how the algorithm assigns destination pointers. Figure 4A illustrates the state of the pointers as in Figure 3, that is with the dest0 and dest2 pointers directed at SLOT3 holding the packet from source 2 and dest1 pointer directed at SLOT4 holding the most recently loaded packet from source 1. This is the status when an incoming packet is newly loaded into the SRAM 3.

The next temporary assignment of destination pointers is illustrated in Figure 4B. The source identifier in the next slot down of the array, SLOT2, is read which identifies source 4 SRC4. From the assignment matrix it is determined that packets from this source are destined for destinations 1 and 2 and therefore the destination pointers dest1, dest2 are realigned to this slot. The destination pointer dest0 is reassigned to null.

The next assignment of destination pointers is shown in Figure 4C. This represents the first part of the final assignment, because the assignment algorithm has reached the end most slot, SLOT1, of the array. This slot holds the packet from source 1 which is destined for destination 1 and therefore the dest1 pointer is assigned to this slot. The dest0 and dest2 pointers are assigned to null.

Figure 4D illustrates the next pass in the final assignment. The source identifier in SLOT2 of the array identifies src 4 as the source which is destined for destinations 1 and 2. There is no point assigning the destination pointer dest2 to this packet because the destination pointer dest1 has already been assigned to the packet which is identified in the SLOT1 of the array and, for bandwidth reasons, the packet should be removed to both destination ports simultaneously. Therefore, no destination pointers are assigned to this slot. In the next slot is a packet from source 2 which is destined for destinations 0 and 2, and so these destination pointers are set for that slot.

After the assignment of destination pointers has been completed, the algorithm controls the SRAM 3 to output the identified packets according to the status of the destination pointers in the packet allocation table.

Figures 5A to 5D illustrate the effect of moving and adding packets into the SRAM 3. Figure 5A shows the status of the packet allocation table 4 in Figure 4D, that is with four slots full, representing four packets in the SRAM 3 and the destination pointers having been finally assigned. Figure 5B shows the effect of adding an additional packet to the SRAM 3. This packet has come in from source 3 and data identifying the packet is added into the vacant slot, SLOT5, of the packet allocation table pointed to by the write pointer wr_pointer. The write pointer is incremented to point to the next vacant slot, SLOT6. The destination pointers remain in place.

Figure 5C shows the state of the packet allocation table when the packet identified in the first slot of the table has been completely transmitted from the SRAM 3 out of its allocated destination port dest1. Note that the write pointer wr_pointer has been decremented to point to the next available slot, SLOT4, and that the assignment of the destination pointer dest1 has been allocated back to null. The next packet to be transmitted from the SRAM 3 is that identified by the data in SLOT2 and this packet is transmitted out of the destination ports dest0 and dest2 as identified by the destination pointers.

Figure 5D shows the effect of removing this packet, i.e. the one identified by the data in SLOT3 in Figure 5A, from the SRAM 3, before the one identified by the data in SLOT1. In this case the data defining the subsequent packet moves up one slot in the array and the destination pointers dest0 and dest2 are reassigned to null.

It will be appreciated that whether or not packets are removed from the SRAM depends on the capability of destinations to receive them. When destinations are capable of accepting data, they return a signal to the TSmerger unit indicating

that they can accept data and then a packet is transmitted. The speed at which packets are removed from the SRAM may also depend on the length of the packets. Generally, each packet stream will contain packets of a common length, although the packet length can differ between different packet streams.

CLAIMS:

1. A stream routing unit for routing each of a plurality of input packet streams to any of a plurality of destinations, the stream routing unit comprising:

a plurality of input ports for receiving respective input streams;

a plurality of output ports associated with respective destinations to which the input packet streams can be routed;

storage means for holding packets of the input packet streams at addressable locations each identifiable by an address;

an assignment data structure identifying for each input stream the or each destination to which it is to be routed; and

a packet allocation data structure holding for each new incoming packet a source identifier identifying the origin of the packet and the address in the storage means where the packet is held, together with information identifying the intended destination of the packet derived from the assignment data structure.

2. A stream routing unit according to claim 1, wherein the input packet streams have a lower bit rate than output streams into which they are merged at the output ports.

3. A stream routing unit according to claim 1 or 2, wherein the assignment data structure is a matrix.

4. A stream routing unit according to any preceding claim, wherein the packet allocation data structure is an array of slots, each slot holding a source identifier and associated address.

5. A stream routing unit according to claim 4, wherein the packet allocation data structure is associated with a write pointer which is configured to point to the next available slot in the array for the source identifier and address of the next incoming packet.

6. A stream routing unit according to claim 4 or 5, wherein information identifying the intended destination of the packet is provided by a set of destination pointers, each associated with a respective output port and each being configured to point to a slot in the array which holds a source identifier and address of a packet intended for that destination associated with that output port.

7. A stream routing unit according to any preceding claim, wherein all packets of a said input stream are of a common length.

8. A data communication system comprising a plurality of packet stream sources each generating a packet stream, a stream routing unit according to any preceding claim and a plurality of destinations for receiving packets of the packet streams generated by the sources.

9. A data communication system according to claim 8, wherein at least one of the destinations comprises a programmable transport interface.

10. A method of routing packet streams from a plurality of sources to any of a plurality of destinations, the method comprising:

receiving said packet streams;

identifying for each input packet stream the or each destination to which it is to be routed using an assignment data structure;

holding each packet of the packet stream at an addressable location identifiable by an address in a storage means;

holding for each new incoming packet a source identifier identifying the origin of the packet and the address in the storage means where the packet is held, together with information identifying the intended destination of the packet derived from the assignment data structure, and using said information identifying the intended destination to route the packet from the storage means to the or each output port associated with the respective identified destination(s).

11. A method according to claim 10, wherein the input packet streams have a lower bit rate than output streams into which they are merged at the output ports.

12. A method according to claim 10 or 11, wherein said information identifying the intended destination of the packet is provided by a set of destination pointers, each associated with a respective output port and each being configured to point to a source identifier and address of a packet intended for the destination associated with that output port.

13. A method according to any of claims 10 to 12, wherein each new incoming packet is held in a packet allocation data structure having a plurality of slots, each slot holding a source identifier and associated address and being associated with a write pointer which is configured to point to the next available slot in the array for the source identifier and address of the next incoming packet.

Abstract

ROUTING OF DATA STREAMS

The routing of data streams is discussed, and particularly routing one or more incoming streams to one or more output destination port. The ability to merge incoming streams is discussed so that several low bit rate input packet streams can be merged into a higher bit rate output stream. An assignment data structure identifies for each input stream the or each destination to which it is to be routed, and a packet allocation data structure holds information about the packets and information about the destination of the packets to allow a memory holding the packets to be controlled accordingly.

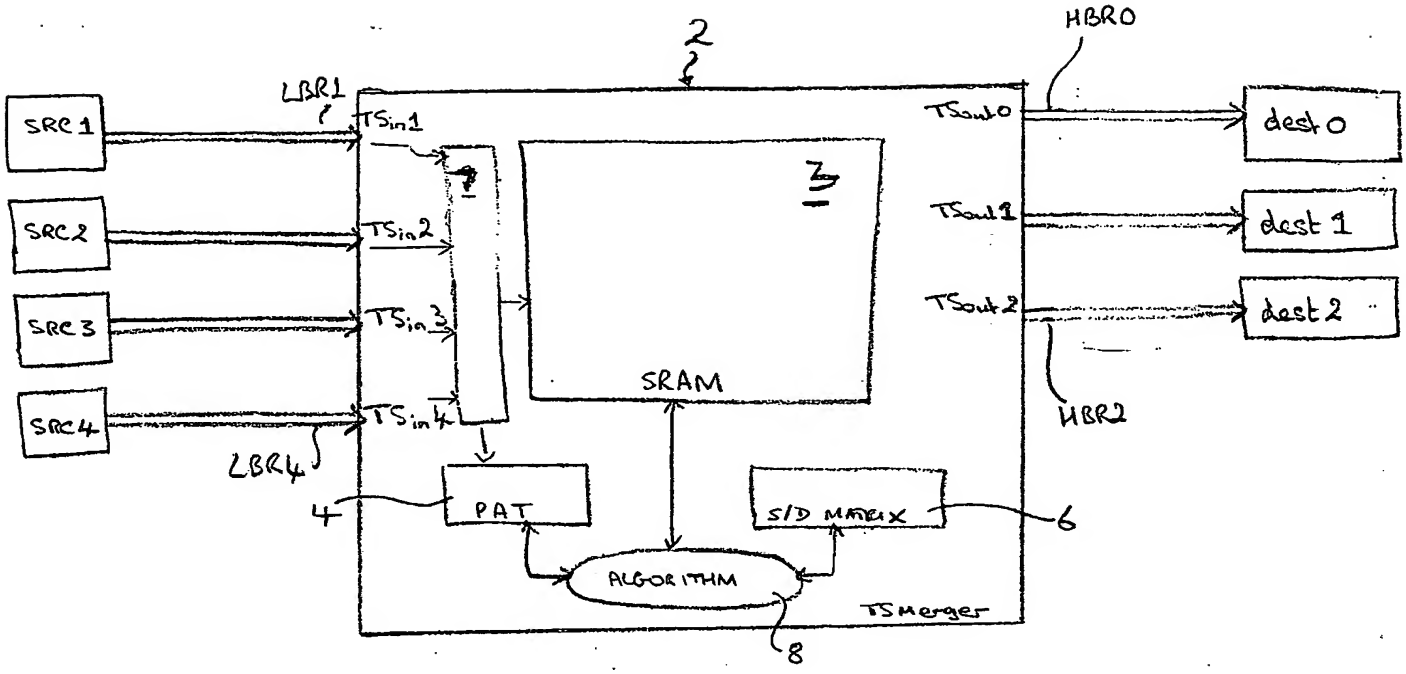


Figure 1

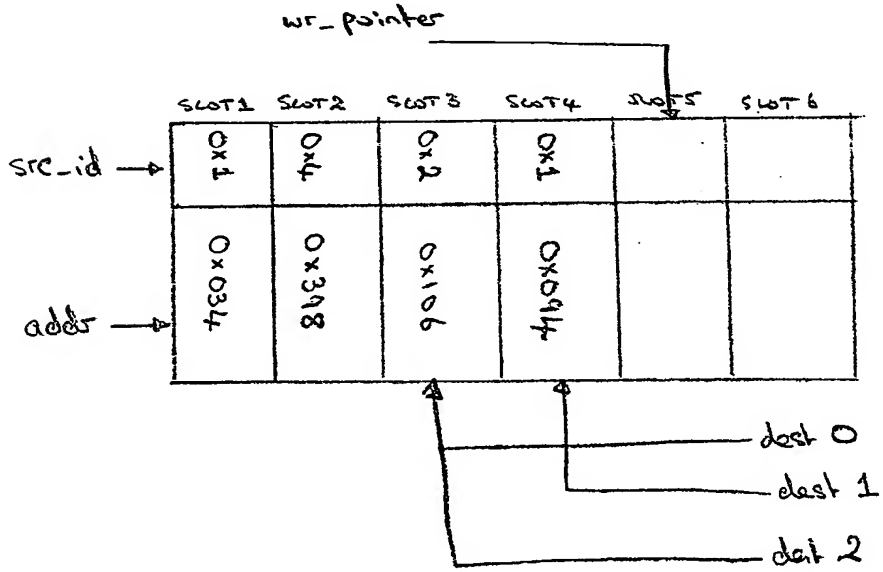


Figure 3

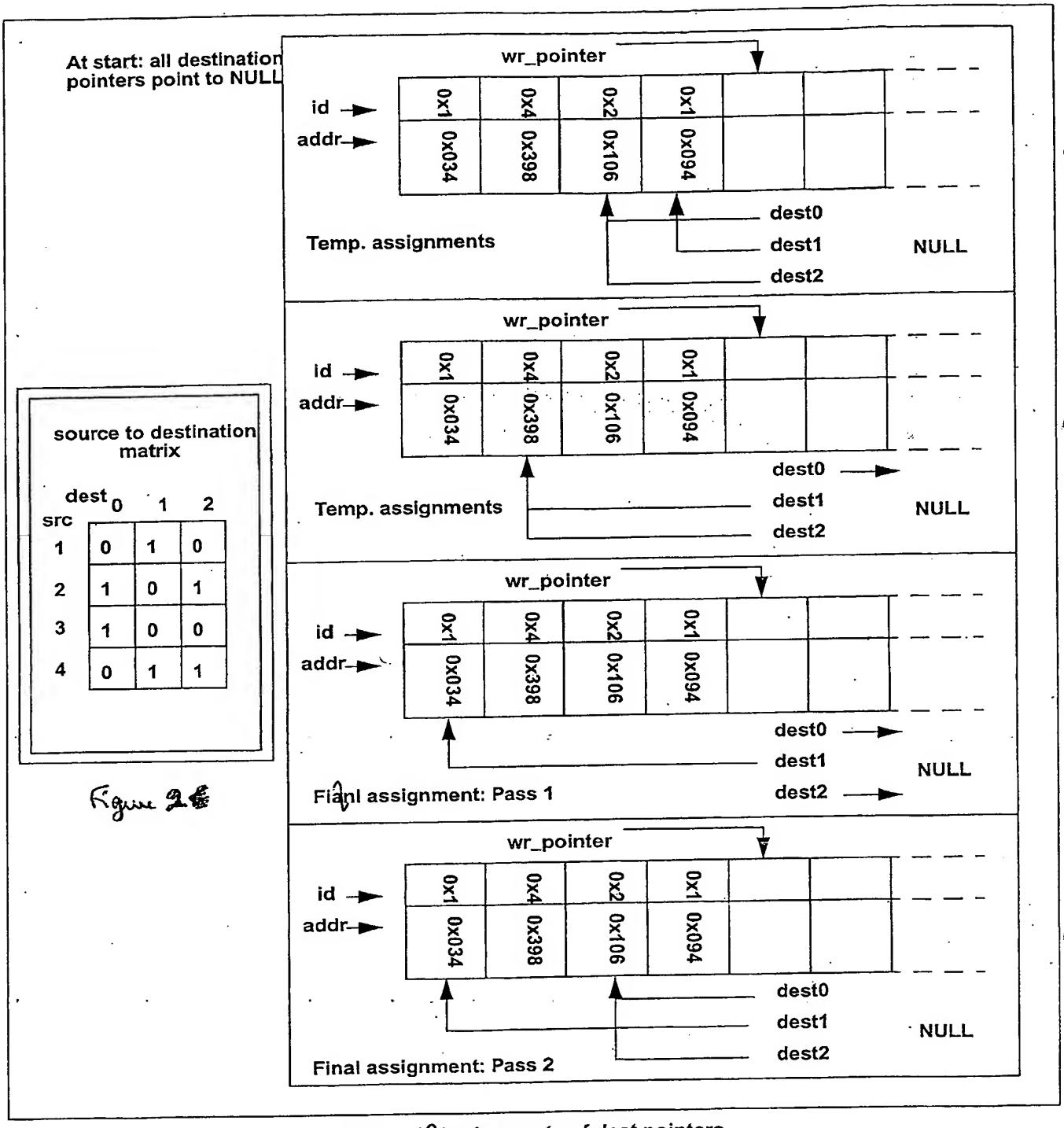
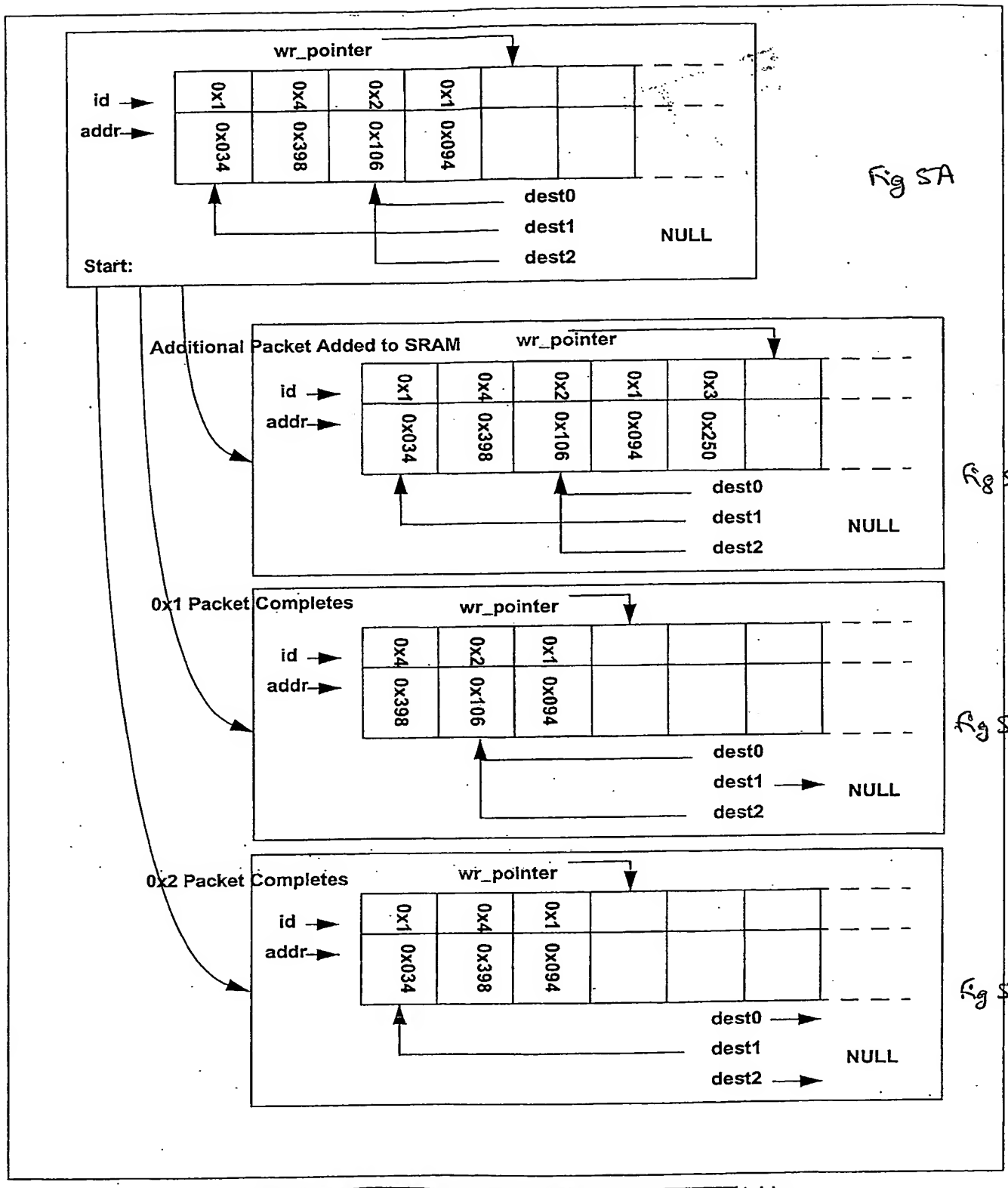


Figure 2B

Figure 4 ~~Assignments of dest pointers~~



~~Figure 2: Adding / Removing Packet Information to/from table~~